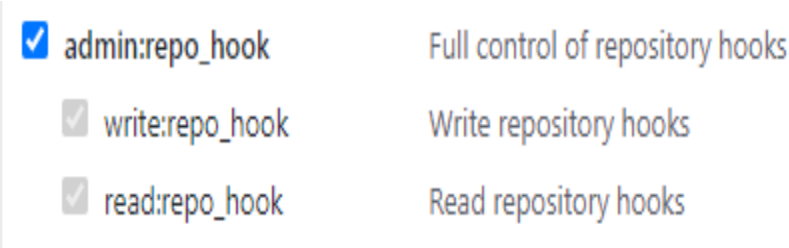
DEV-OPS PIPELINE FORCE

INSTRUCTIONS TO USE

PRE-REQUISITES:

- Revature Bastion connection .pem file & password
- Be provided with PipelineDocs.zip
- Jenkinsfile templates - Angular & Maven files (PipelineDocs.zip)
- Powershell or Bash console to SSH with
- On Github account settings - navigate to DeveloperSettings -> PersonalAccessTokens -> Create a new personal access token and select the following option only:



FOR ANGULAR PROJECTS

- SSH into Bastion EC2
- SSH into the Jenkins EC2
- Execute 'docker rmi -f $(docker images -q)'
    - This deletes the EXTREMELY large files from the storage thus freeing up resources and preventing Jenkins jobs from failing due to memory/space issues.

DIRECTIONS:

1. (TRAINEE) - Fork the selected repository into personal GitHub
2. (TRAINEE) - Edit the appropriate Jenkinsfile
    a. RegisterFilename - change this from the list above line 12 using these options listed.
    b. Service (MAVEN ONLY) - Change this from the list using the names in lines 23~27

```
11          //caliber-batch
12          RegisterFilename = "CHANGEME"
13          //THE NAME OF THE DOCKER ECR REPO
```

(MAVEN PROJECT ONLY)

```
27          //caliber-quality-audit-service
28          Service ="caliber-quality-audit-service"
29      }
```

3. (TRAINEE) - Add the Jenkinsfile to the project root. Commit + push to Git repository
4. (TRAINER) - SSH into the Bastion server with Revature credentials
5. Switch to '/home/august.duet/pipelineforce'
6. execute './initialize_pipeline' this will run the .bat script to invoke the Ansible Playbook to set everything up.
7. (PROMPT) Enter Github Username:
8. (PROMPT) Enter Github API Token:
9. (PROMPT) Enter Full Repo URL (including .git):
10. (PROMPT) Enter branch name

CONGRATULATIONS!! Your automated pipeline is now being configured by Ansible, Jenkins job is being built automatically and configuring all the parameters to handle any future Git Pushes/Merges/Tag-Pushes.

PIPELINE OVERVIEW

Phases:

1. Fork & Configuration by Trainee
2. Trainer Ansible - Invoke from Bastion
3. Ansible -> Setup Jenkins Job on server, configure webhook and build the job.

AUTOMATED PIPELINE

1. Any Git push/merge/tag-push will trigger the webhook in Jenkins and start the automated pipeline process.
2. Jenkins -> Will run a clean package lifecycle (skipping tests) to produce artifact (.jar/.war/.ear)
3. Jenkins -> Builds the Docker image with a tag 'latest'
4. Jenkins -> Pushes the image to the AWS ECR Repo
5. Jenkins -> Cleans Docker images from Jenkins server freeing up resources
6. (ON FAILURE) - Jenkins will send off email notification to centerofexcellence@revature.com
7. (ON ECR PUSH) - Spinnaker has a webhook to trigger Spinnaker pipeline
8. Spinnaker -> Pulls Helm charts from AWS S3 storage
9. Spinnaker -> Uses Helm charts in the 'Bake' stage to build JSON artifacts for Rollbacks & Version Control in Spinnakers Deployment phase.